


```

LL          IIIIII          SSSSSSSS
LL          IIIIII          SSSSSSSS
LL          II             SS
LL          II             SS
LL          II             SS
LL          II             SS
LL          II             SSSSSS
LL          II             SSSSSS
LL          II             SS
LL          II             SS
LL          II             SS
LL          II             SS
LLLLLLLLLLLL IIIIII          SSSSSSSS
LLLLLLLLLLLL IIIIII          SSSSSSSS

```

LIE
LIE
LIE


```
1 0001 0 MODULE LIB_SUBS ( ! General subroutines for LIBRARIAN
2 0002 0
3 0003 0 LANGUAGE (BLISS32),
4 0004 0 IDENT = 'V04-000'
5 0005 0 ) =
6 0006 1 BEGIN
7 0007 1
8 0008 1 *****
9 0009 1 *
10 0010 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
11 0011 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
12 0012 1 * ALL RIGHTS RESERVED.
13 0013 1 *
14 0014 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
15 0015 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
16 0016 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
17 0017 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
18 0018 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
19 0019 1 * TRANSFERRED.
20 0020 1 *
21 0021 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
22 0022 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
23 0023 1 * CORPORATION.
24 0024 1 *
25 0025 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
26 0026 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
27 0027 1 *
28 0028 1 *
29 0029 1 *****
30 0030 1
31 0031 1 ++
32 0032 1
33 0033 1 FACILITY: Library command processor
34 0034 1
35 0035 1 ABSTRACT:
36 0036 1
37 0037 1 The VAX/VMS librarian is invoked by DCL to process the LIBRARY
38 0038 1 command. It utilizes the librarian procedure set to perform
39 0039 1 the actual modifications to the library.
40 0040 1
41 0041 1 ENVIRONMENT:
42 0042 1
43 0043 1 VAX native, user mode.
44 0044 1
45 0045 1 --
46 0046 1
47 0047 1
48 0048 1 AUTHOR: Benn Schreiber, CREATION DATE: 21-June-1979
49 0049 1
50 0050 1 MODIFIED BY:
51 0051 1
52 0052 1 V01.01 Benn Schreiber 8-Nov-1979
53 0053 1 Add routine getfilnamdesc
54 0054 1 --
55 0055 1
56 0056 1
```

LIB SUBS
V04=000

H 7
16-Sep-1984 02:04:43
14-Sep-1984 12:38:11

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[LIBRAR.SRC]SUBS.B32;1 Page 2 (2)

```
: 58      0057 1 LIBRARY
: 59      0058 1 'SYSS$LIBRARY:STARLET';
: 60      0059 1 REQUIRE
: 61      0060 1 'PREFIX';
: 62      0244 1
: 63      0245 1 EXTERNAL ROUTINE
: 64      0246 1 lib$get_vm : ADDRESSING_MODE (GENERAL), !Allocate virtual memory
: 65      0247 1 lib$free_vm : ADDRESSING_MODE (GENERAL);!Deallocate virtual memory
```

```

: 67      0248 1 GLOBAL ROUTINE lib_get_mem (bytes, retadr) =
: 68      0249 1 |
: 69      0250 1 | allocate virtual memory
: 70      0251 1 |
: 71      0252 1 lib$get_vm (bytes, .retadr):

```

```
.TITLE      LIB SUBS
.IDENT      \V04-000\

.EXTRN      LIB$GET_VM, LIB$FREE_VM

.PSECT      $CODE$,NOWRT,2

.ENTRY      LIB_GET_MEM, Save nothing
PUSHL      RETADR
PUSHAB     BYTES
CALLS      #2, LIB$GET_VM
RET
```

```
; Routine Size: 16 bytes,    Routine Base: $CODE$ + 0000
```

```

: 72      0253 1 GLOBAL ROUTINE lib_free_mem (bytes, address) =
: 73      0254 1 |
: 74      0255 1 | deallocate virtual memory
: 75      0256 1 |
: 76      0257 1 lib$free_vm (bytes, address);

```

```

.ENTRY    LIB_FREE_MEM, Save nothing                ; 0253
PUSHAB    ADDRESS                                   ; 0257
PUSHAB    BYTES
CALLS     #2, LIB$FREE_VM
RET

```

```
; Routine Size: 16 bytes,    Routine Base: $CODE$ + 0010
```

```

: 77      0258 1 GLOBAL ROUTINE lib_get_zmem (bytes, retadr) =
: 78      0259 2 BEGIN
: 79      0260 2 |
: 80      0261 2 |   Get zeroed memory
: 81      0262 2 |
: 82      0263 2 perform (lib_get_mem (.bytes, .retadr));
: 83      0264 2 CH$FILL (0, .bytes, ..retadr);
: 84      0265 2 RETURN true
: 85      0266 1 END;

```

```
003C 00000 .ENTRY LIB_GET_ZMEM, Save R2,R3,R4,R5 ; 0258
```


LIB SUBS
V04=000

J 7
16-Sep-1984 02:04:43
14-Sep-1984 12:38:11

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[LIBRAR.SRC]SUBS.B32;1 Page 4
(3)

04	AC	00	D6	7E	04	AC	7D	00002	MOVQ	BYTES, -(SP)	:	0263
				AF		02	FB	00006	CALLS	#2, LIB_GET_MEM	:	
				0E		50	E9	0000A	BLBC	STATUS, 1\$	:	
				50	08	BC	D0	0000D	MOVL	@RETADR, R0	:	0264
				6E		00	2C	00011	MOVCS	#0, (SP), #0, BYTES, (R0)	:	
						60		00017			:	
				50		01	D0	00018	MOVL	#1, R0	:	0265
						04	0001B	1\$:	RET		:	0266

; Routine Size: 28 bytes, Routine Base: \$CODE\$ + 0020

; Routine Size: 189 bytes, Routine Base: \$CODE\$ + 003C

LIB SUBS
V04=000

M 7
16-Sep-1984 02:04:43
14-Sep-1984 12:38:11

VAX-11 Bliss-32 V4.0-742
DISK\$VMMASTER:[LIBRAR.SRC]SUBS.B32;1 Page 7 (4)

```
145 0324 1 GLOBAL ROUTINE getfilnamdesc (fab, filedesc) =
146 0325 2 BEGIN
147 0326 2 ++
148 0327 2 FUNCTIONAL DESCRIPTION:
149 0328 2
150 0329 2 This routine returns a string descriptor for a file.
151 0330 2
152 0331 2 Inputs:
153 0332 2
154 0333 2 fab Address of the fab
155 0334 2 filedesc Address of string descriptor to store into
156 0335 2
157 0336 2 Outputs:
158 0337 2
159 0338 2 filedesc is filled in. The strings tried are:
160 0339 2
161 0340 2 1) resultant name string
162 0341 2 2) expanded name string
163 0342 2 2) filename string
164 0343 2
165 0344 2 --
166 0345 2
167 0346 2 MAP
168 0347 2 fab : REF BBLOCK,
169 0348 2 filedesc : REF BBLOCK;
170 0349 2
171 0350 2 BIND
172 0351 2 nam = .fab [fab$l_nam] : BBLOCK;
173 0352 2
174 0353 2 LOCAL
175 0354 2 nameptr;
176 0355 2
177 0356 2 IF (filedesc [dsc$w_length] = .nam [nam$b_rsl]) NEQ 0 !If resultant name present
178 0357 2 THEN filedesc [dsc$a_pointer] = .nam [nam$l_rsa]
179 0358 2 ELSE IF (filedesc [dsc$w_length] = .nam [nam$b_esl]) NEQ 0 !If expanded name present
180 0359 2 THEN filedesc [dsc$a_pointer] = .nam [nam$l_esa]
181 0360 2 ELSE BEGIN
182 0361 2 filedesc [dsc$w_length] = .fab [fab$b_fns]; !Use filename string
183 0362 2 filedesc [dsc$a_pointer] = .fab [fab$l_fna]; ! if all else fails
184 0363 2 END;
185 0364 2
186 0365 2 ! Allocate memory and copy the filename to it
187 0366 2
188 0367 2 lib get_mem (.filedesc [dsc$w_length], nameptr);
189 0368 2 CH$MOVE (.filedesc [dsc$w_length], .filedesc [dsc$a_pointer], .nameptr);
190 0369 2 filedesc [dsc$a_pointer] = .nameptr;
191 0370 2 RETURN true
192 0371 1 END;
```

!Of getfilnamdesc

```
007C 00000
5E 04 C2 00002
51 04 AC 7D 00005
50 28 A1 D0 00009
```

```
.ENTRY GETFILNAMDESC, Save R2,R3,R4,R5,R6
SUBL2 #4, SP
MOVQ FAB, R1
MOVL 40(R1), R0
```

```
: 0324
:
: 0351
:
```


LIB SUBS
V04=000

B 8
16-Sep-1984 02:04:43
14-Sep-1984 12:38:11

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[LIBRAR.SRC]SUBS.B32;1 Page 9
(5)

56	04	A2	9E	0000D	MOVAB	4(R2), R6	0357
62	03	A0	9B	00011	MOVZBW	3(R0), (R2)	0356
		06	13	00015	BEQL	1\$	
66	04	A0	D0	00017	MOVL	4(R0), (R6)	0357
		14	11	0001B	BRB	3\$	
62	0B	A0	9B	0001D	MOVZBW	11(R0), (R2)	0358
		06	13	00021	BEQL	2\$	
66	0C	A0	D0	00023	MOVL	12(R0), (R6)	0359
		08	11	00027	BRB	3\$	
62	34	A1	9B	00029	MOVZBW	52(R1), (R2)	0361
66	2C	A1	D0	0002D	MOVL	44(R1), (R6)	0362
		5E	DD	00031	PUSHL	SP	0367
		62	3C	00033	MOVZWL	(R2), -(SP)	
00	BE	FECC	CF	02	CALLS	#2, LIB GET MEM	
		B6	62	28	MOV3	(R2), @0(R6), @NAMEPTR	0368
		66	6E	D0	MOVL	NAMEPTR, (R6)	0369
		50	01	D0	MOVL	#1, R0	0370
			04	00047	RET		0371

; Routine Size: 72 bytes, Routine Base: \$CODE\$ + 00F9

; 193 0372 1 END
; 194 0373 0 ELUDOM

PSECT SUMMARY

Name	Bytes	Attributes
\$CODE\$	321	NOVEC,NOWRT, RD , EXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)

Library Statistics

File	Total	Symbols Loaded	Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[SYSLIB]STARLET.L32;1	9776	37	0	581	00:01.0

COMMAND QUALIFIERS

; BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS\$:SUBS/OBJ=OBJ\$:SUBS MSRC\$:SUBS/UPDATE=(ENH\$:SUBS)
; Size: 321 code + 0 data bytes

LIB SUBS
V04=000

C 8
16-Sep-1984 02:04:43

VAX-11 Bliss-32 V4.0-742

Page 10

: Run Time: 00:08.7
: Elapsed Time: 00:18.3
: Lines/CPU Min: 2566
: Lexemes/CPU-Min: 28176
: Memory Used: 86 pages
: Compilation Complete

_S

Sy

--

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

LIB

0202

AH-BT13A-SE
 VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY